Cross Streets: Microcomputers and Amateur Radio, Part 1

A Byte of History Shows Just How Far We've Come in 40 Years

By Cory GB Sickles, WA3UVV, <wa3uvv@gmail.com>

Have you ever wondered how computers, microprocessors and the ever-present microcontroller became such staples of amateur radio? Do you wish you knew more about what goes on inside those hunks of silicon and how you could use them to do things around the shack? Are you one of those people who likes to learn new things? Of course you are. *You wouldn't be in amateur radio if that were not the case.*

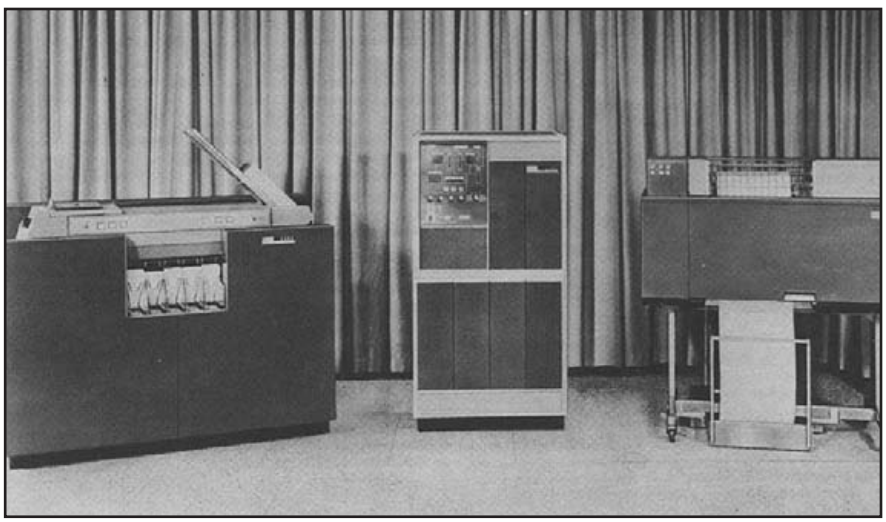


Photo A. We've come a long way since the massive IBM 1401 computer. Across the timeline, systems got smaller, faster, more efficient, and *cheaper.* *(Courtesy of Wikimedia Commons)*

About This Series

In Part I of this *WorldRadio Online* series, we'll take you back to the origins of the microcomputer, touch on some machines of the past and bring us to the state-of-the-art processors that appear in everything from transceivers to tuners to keyers, *and more.*

Of course, you're encouraged to research topics on your own for more in-depth information, as there's only so much room for a piece of this scope. Also, as much as I've tried to present this in a linear and chronologically ordered way, I am a *non-linear thinker* and many things intertwine or overlap.

So, don't take Part I as a definitive *on this day in history* timeline coverage of the evolution of microcomputers. While it's not *that,* I hope it will be informative and enjoyable both to those who lived through these times and to new radio amateurs who have no idea what we're talking about. In short: *Be curious and look up the things you don't immediately recognize. Who knows, it just might make you a more interesting person.*

Back in the Day . . .

It may be difficult for some to imagine amateur radio without the influence of microprocessors and personal computers. However, there was a time when computers were *way more* expensive than a house and the only amateur radio applications might be if some lucky person had a little extra time on his or her employer's system to program propagation predictions or beam headings in assembly language or FORTRAN IV.

Take a look at old pictures of an IBM 1401, **Photo A**, and you'll have some idea of the massive size of these machines. Memory — known as *core* — was made of arrays of wires running through miniature toroidal cores, as each bit was stored magnetically. At least they were solid state. *The first generation of systems used vacuum tubes!*

As time progressed, things got smaller and a new type of system — the *minicomputer* — was born. The Central Processing Unit (CPU) was a collection of transistors and diodes that handled the various functions that make a computer *a computer.*

Eventually, solid-state Random Access Memory (RAM) replaced the magnetic core. Companies like Digital Equipment Corporation and Data General would enter the field as systems became more affordable. Price tags in the millions of dollars were now in the hundreds of thousands and would soon be in the tens of thousands. The DEC PDP-8 was a leader in breaking the sub-$20,000 price-point barrier.

Enter Intel's 4004 and 8008

In 1971, a fairly young company by the name of Intel announced a product that would create ripples across the electronics industry for generations to come, while offering some in the industry the opportunity to become billionaires and force others into bankruptcy. The device was the 4004 microprocessor.

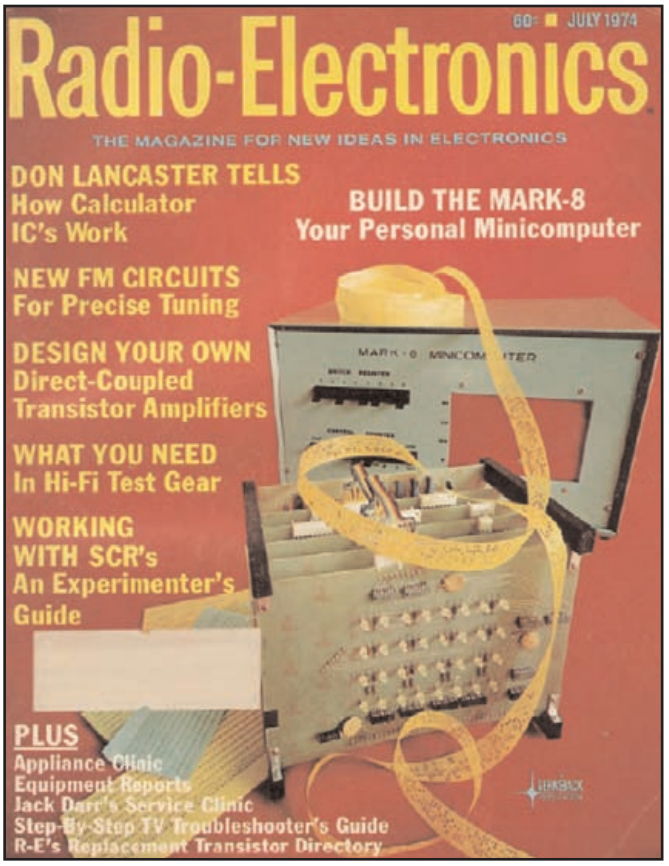


Photo B. Many computer historians point to the July 1974 edition of *Radio-Electronics* as the *official* start of the microcomputer revolution. *(Courtesy of KI6SN)*

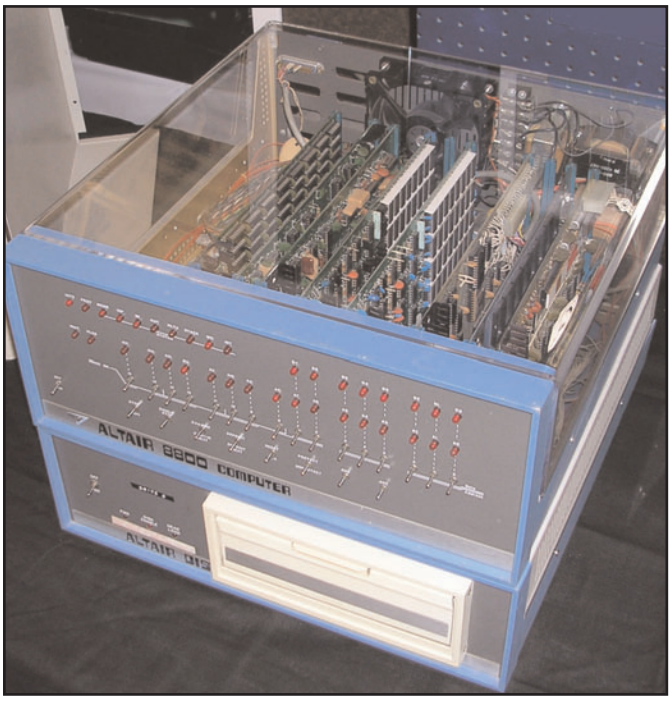


Photo C. With the mid-1970s introduction of the Altair 8800 Minicomputer kit, writes WA3UVV, the computer hobby market "really started to take off." *(Courtesy of Swtpc6800 via Wikimedia Commons)*

Photo D. The January 1975 issue of *Popular Electronics* magazine featured an article about the Altair 8800 Minicomputer kit, and the rest — as they say — is history. *(Courtesy of Wikimedia Commons)*

This 16-pin integrated circuit was the foundation for early programmable devices, calculators and the first microcomputers — although that term hadn't been introduced, just yet. The 4004 was a 4-bit processor and couldn't do much as a general purpose device, but soon the 8008 was announced and the concept of the hobby computer began.

Initial versions of the 8008 could run with a clock speed (frequency) of 500 kHz. You read that right: *Half a megahertz.*

Made up of 3,500 transistors, this 8-bit processor had a rudimentary instruction set and could access up to 16 kilobytes (KB) of RAM. My first computer was built around one of these, on a wire wrapped board with toggle switches and lights for address, data, and status indicators. How much memory did it have? How about 256 bytes. *Not "giga," not "mega," not "kilo," just plain old bytes.*

How useful was it? Well, you could manipulate some numbers as a binary calculator, control things to be on or off based on toggle switch positions or some calculation, and just make the lights blink in various sequences. At one point, I also programmed it to generate CW. In essence, it was my first memory keyer.

*(**NOTE:** "Radio-Electronics," a well-known magazine at the time, published an article in its July 1974 issue, **Photo B**, about the Mark-8 Minicomputer, which was 8008-based. While mostly a set of diagrams and plans that required some scrounging for parts, many computer historians point to this as the "official" start of the microcomputer revolution. – WA3UVV)*

It had no way to store or save a program permanently. If power was cut, it was back to reprogramming it via the switch-

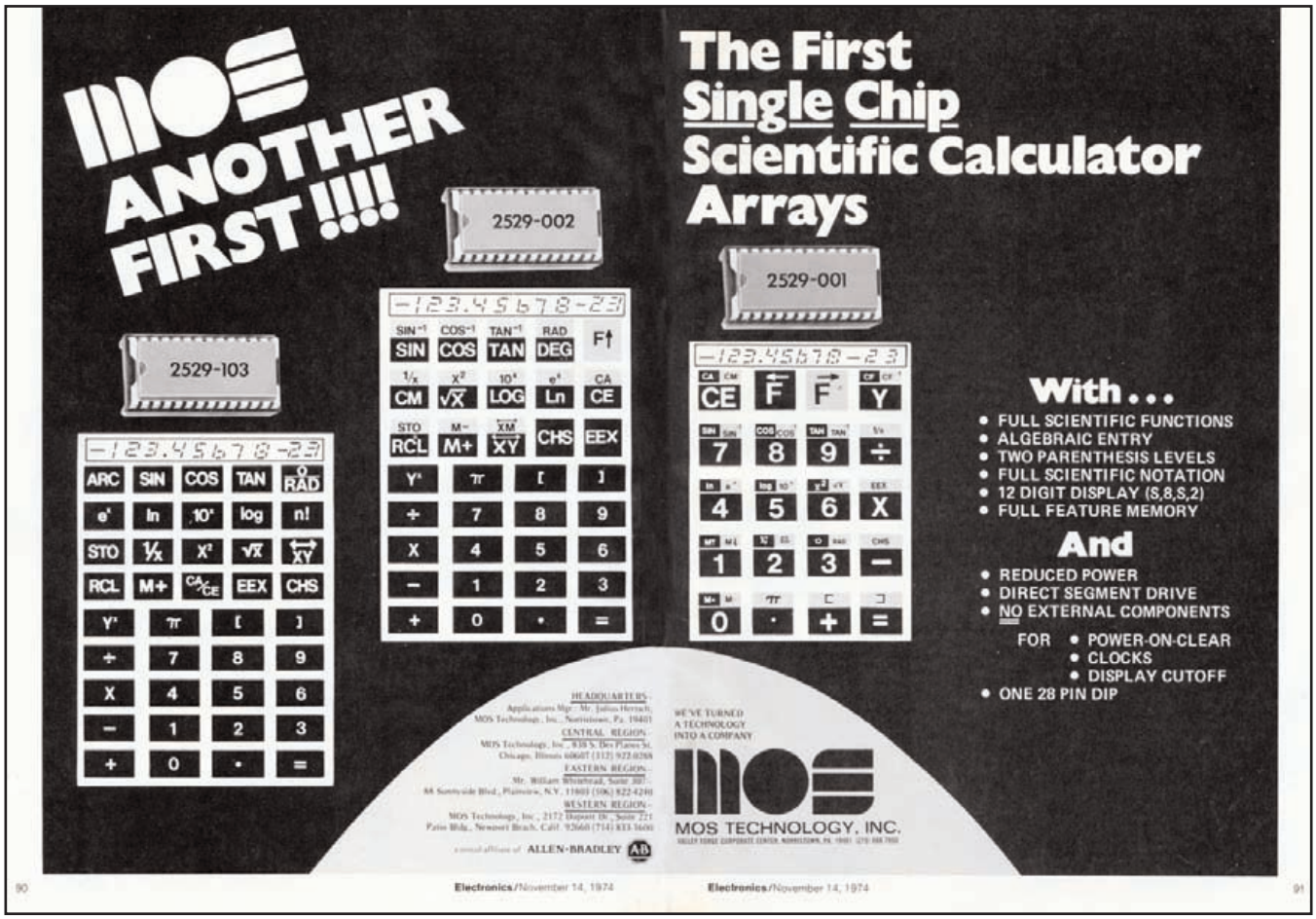


Photo E. MOS Technology introduced the "first single-chip scientific calculator arrays," as trumpeted in this 1974 advertisement. *(Courtesy of Wikimedia Commons)*

es, one byte at a time. I wish I could say I still have that Single Board Computer (SBC), but when the 8080 — a vastly improved 8-bit processor came on the market, I ended up abandoning the 8008 and used many of the parts to build an 8080 SBC with a whopping 1 KB of memory.

The Computer Hobbyist

The January 1975 issue of *Popular Electronics* magazine featured an article about the Altair 8800 Minicomputer kit — and the hobby market really started to take off, **Photos C** and **D**.

As a side note, microprocessors were very expensive at this time. However, Intel made the 8080 chipset available as a *Student Special* package, complete with several pounds of documentation for a very reasonable price. Many of us cash-strapped kids grabbed at the opportunity.

By now, there were a small number of computer kits on the market and various clubs, newsletters, and even computer magazines were a reality. One of the most active subsets of early microcomputer experimenters were radio amateurs, of course. We had the electronics background, knew which end of a soldering iron to hold and weren't afraid of a little *pop-n-smoke* from time to time.

To be sure, this was still an era when tubes ruled when it came to our transceivers, but many of those who had been experimenting with solid-state devices were fairly comfortable. Learning new terms like DMA, Write Cycle, and Buss Latency were just part of the fun.

As kits (typically CPU, 1 KB or less of RAM, motherboard for expansion, power supply, and case) made inroads, other companies entered the fray with microprocessors of their own. Soon the 8080 was joined by the 1600, 1802, 2650, 6502, 6800, F-8, SC/MP, Z-80, and others. While many failed to gain a huge following and have disappeared into obscurity, there were a few notables along the way.

The Zilog Z-80

The Z-80, manufactured by Zilog, was an 8080 upgrade. It could run the same binaries with increased speed, had an additional set of registers and one important hardware feature — it could handle the refresh cycle required by dynamic RAM without external hardware. This was *a big* advantage, since dynamic RAM was cheaper and denser than static RAM. As you might imagine, a lot of 8080 designs were soon upgraded to Z-80s.

RCA Develops CMOS Processor

RCA developed the first Complementary Metal Oxide Semiconductor (CMOS) processor — CDP1802 as a single-chip follow up to the 1801/1810 chipset. CMOS chips have the advantage of using much less power and can run with a wider variety of voltages. Also, the frequency can be easily varied and even brought to zero.

The 1802 found use in the aerospace industry and was used in space probes like Galileo and some OSCAR designs, because

Photo F. The MOS Technology KIM-1 microcomputer was priced at $245 in this advertisement from May 1976. *(Courtesy of Wikimedia Commons)*

there were versions that were "hardened" against radiation. RCA also developed supporting chips for inexpensive video displays and sound generation, plus a proprietary disk operating system (DOS).

The Intersil 1600, R.I.P.

The Intersil 1600 was another CMOS processor brought to market. It was promoted as a low-power foundation for micros that could fully emulate the DEC PDP-8. This 12-bit chip had the promise of being able to use a library of readily available software already written for the best-selling minicomputer at the time. *(**NOTE:** See "How DEC's 'Straight 8' Became a Crowd Favorite" – WA3UVV)*

While systems like TLF's "Mini-12" were good hardware options, their marketing left much to be desired. Intersil's own Intercept Jr. was limited by design to be an evaluation system and not much else. Others designed systems using similar I/O as the DecMate, which was incompatible with the previous PDP-8 line. Sadly, this lesson would have to be learned later by others, when the IBM-PC and a gallery of alleged "compatibles" hit the market. Sad to say, the 1600 died an early death and took a number of deluded hostages with it. *(**NOTE:** All hope is not lost, however. SpareTimeGizmos.com offers a current and "fully compatible" SBC of the PDP-8. I'm in the middle of building one. – WA3UVV)*

No Frills: Motorola 6800

Motorola came up with a fairly straightforward design in the 6800. This 8-bit processor was supported by a collection of additional chips for serial and parallel I/O, plus received a lot of attention and marketing push from a large and well-established manufacturer. Southwest Technical Products, whose specialty had been in the audio industry up to this point, created a line of kits and peripherals based on the 6800 and its friends, plus some new bus designs — the SS-50 and SS-30.

Coming to Terms

You may be asking: *What's a bus?* Basically, it's just a standard way — *and I use those words loosely* — of interconnecting circuit boards such as CPU front panel, memory, I/O, etc. in such a way that everything had a uniform pin-out, to allow for interchangeability of options and assemblies.

When the 8080 kits came on the market, Altair was the first to offer a standardized bus — the Altair bus. Later, this became known as the S-100 bus — a clue to the number of pins — and much later, gained formalization as the IEEE-696.

A popular phrase, "The great thing about standards is that there are so many" certainly applies to the S-100 bus. While most boards *played well* with other boards, certain problems arose when a number of pins were initially unassigned. Thus, one manufacturer might use this "extra" pin to communicate between boards it made, while someone else would use it for an alternative purpose. Those uses were typically incompatible. *(**NOTE:** Pin 67 is a notorious example of some undefined or ill-defined pins on the S-100 bus. – WA3UVV)*

The Price Is Right? Ask Chuck Peddle!

Going back to the 6800, microprocessors were still on the pricey side for the most part. The 6800 was available in single-quantity pricing for about $300 initially. Companies try to recoup their R&D costs as quickly as possible, and Motorola was no exception.

For reference, $300 in mid-'70s money adjusts to about $1,200 in current cash. *Yeah, not exactly cheap.* Of course, that $600 price tag for a mid-'70s Kenwood TS-520 may not seem so attractive right now, either. See how much more functionality your dollar buys in the way of amateur radio gear today?

Potential customers of the 6800 thought the price a bit steep, if they were going to find ways to utilize it in their products. When pressed, many suggested pricing of about one-tenth as much. A key designer at Motorola — Chuck Peddle, <http://bit.ly/RRYQDo> — took notice of this and along with a few of his coworkers started a rogue project to reduce the cost associated with the 6800.

While Peddle's name may not be familiar, he remains one of the lesser-known heroes in the microcomputer world as his inspiration and drive created a foundation that would be capitalized upon by Steve Jobs and — more importantly — Steve Wozniak, who was a radio amateur at one time.

When Peddle showed his supervisors at Motorola what he and his team were working on, they were not impressed. The company felt that the 6800 was the greatest thing since indoor plumbing and should stay priced at $300. Chuck and a few other designers decided there was a future in what they were doing and decided to start their own venture. Thus, MOS Technology — which was a low-cost chip fabricating company with no real design capabilities — entered the picture, **Photo E**.

Soon after, the 6501 was born. This chip was a pin compatible processor to the 6800 and offered similar opcodes. Because of MOS Technology's manufacturing facilities, they were able to offer it at the price-point that other companies wanted. It generated a feeding frenzy.

Photo G. Designed for engineers working in industry, the KIM-1 included the 6502, 1 KB of RAM (expandable on board to 5K), cassette and serial I/O, expansion connectors, a 6-digit LED display, and keypad. *(Courtesy of Wikimedia Commons)*

When Motorola heard about the 6501 and realized who was behind it, they did what any big company would do — they deployed their legal strike force and attacked.

Long story short: Both parties settled and it was agreed that all the 6501s made would be destroyed. I'd be willing to bet there's still one or two out there as a piece of history, though. MOS Technology quickly did a redesign and announced the 6502 and its low-low price of $25. *(**NOTE:** Literally overnight, the MSRP of other processors fell to line up with the 6502! – WA3UVV)*

Although it was now *mostly like* the 6800, it was different enough to be "lawyer proof" and was no longer pin compatible. MOS Technology needed to get these low-cost chips into the hands of designers as quickly and inexpensively as possible. They created an SBC known as the KIM-1, **Photos F**, **G**, and **H**.

Designed for engineers working in industry, the KIM-1 sold for $245 and included the 6502, 1 KB of RAM (expandable on board), cassette and serial I/O, expansion connectors, a 6-digit LED display, and keypad. Very quickly, the hobbyist market's demand for the KIM-1 outpaced the intended market and another *hit product* was gaining support from users.

Photo H. "Radio amateurs really took notice of the KIM-1," writes WA3UVV. "It was well covered in *73* magazine and other Wayne Green, W2NSD/1, publications, such as *Kilobaud.*" *(Courtesy of KI6SN)*

KIM-1 Gets Some Ink

Radio amateurs really took notice of the KIM-1. It was well covered in *73* magazine and other Wayne Green, W2NSD/1, publications, such as *Kilobaud.* A prolific publisher, I think he once had more than half a dozen computer-related magazines going at the same time, plus a software distribution business, and more.

*(**NOTE:** Many of us fondly remember a time when "73" was a huge magazine,*

filling scores of pages each issue. If you look closely, you'll see that advertising made that possible and a good portion of the advertising was microcomputer related. – WA3UVV)

With significant coverage in amateur radio circles, more micros were sold, more software written and more peripherals announced — all driving additional coverage. New products were introduced every month by companies you'd previously never heard of. *It was a heady time, indeed.*

A Life Changing Encounter

By this point, I'd already built a few single-board systems and was well entrenched in the Intel/Zilog chips and didn't feel the need to learn something new. But a visit to a friend in my hometown radio club changed that.

Tim Soxman, W3ZVT (SK), was a fan of the KIM-1. Many know of Tim from his exceptionally machined straight keys. *(**SEE:** An example of W3ZVT's handiwork at <http://www.mtechnologies.com/soxman/>. – WA3UVV)*

I knew him as an evangelist for the 6502 and KIM-1. In less than an hour, he convinced me that I should start playing with the 6502. I wish I still had my original KIM-1, but thanks to another designer of SBC's and peripherals, I can relive the KIM today. The Micro-Kim keeps the KIM-1 alive, along with an updated Apple I design and an 8080-based system, with real toggle switches and more at: <http://www.briel computers.com/>.

Whew! . . . And What's Ahead

I've covered a lot of history to give you a foundation on what's to come. In Part II, more about the resurgence of the KIM-1 and other old machines, the incredibly long life of the 6502, more serious computers with standard operating systems, and the emergence of the Apple II, TRS-80, PET 2001, plus the explosion of amateur radio applications that followed.

Your questions and comments are certainly welcome. Write to me at: <wa3uvv@arrl.net>. – WA3UVV.

How DEC's 'Straight 8' Became a Crowd Favorite

By Cory GB Sickles, WA3UVV

The DEC PDP-8 — released in the mid-'60s — was an exceptional value for all it could do. The affordability of the original "Straight 8" and subsequent versions made for a long legacy of systems that were still selling well in the late '70s.

As many have expounded on the value of Reduced Instruction Set Computer (RISC) architectures, the PDP-8 remains the epitome of RISC. With only 8 basic instructions, it utilized its 12-bit design as a highly efficient model — conserving memory within its original 4-KW constraints. It is an excellent way to learn about how computers work "where the rubber meets the road" and how to compose efficient software — the antithesis of Windows.

DEC lost out on an incredible opportunity with the Intersil 6100. While it was focused on the chip as a compact and low-cost solution to future PDP-8 and DecMates, its tunnel vision kept it from realizing the rich potential of the emerging microcomputer market. In addition, it considered, but declined to release lower-cost versions of its language and operating system software for such systems.

With the overall lack of entry-level systems that could be easily expanded, plus the absence of reasonably priced software, the 6100 was never able to be embraced by hobbyists — many of whom ended up writing software and integrating systems for specific business purposes. The PDP-8 was the best-selling computer until eventually being surpassed by sales of the Apple II.

Today, you can either try to find, *and pay dearly for,* an original PDP-8 or versions such as the /M, /I, /L, or /A. Or you can purchase an SBC-6120 kit from Spare Time Gizmos <http://sparetimegizmos.com> — utilizing the Harris 6120, an improved version of the 6100. It can give you the opportunity to work with the PDP-8 architecture, toggle switches, and more.

Through the efforts of many angels that keep the '8 alive, you can find access to software and more via something that was also in its infancy during the '70s — *the Internet.*

Photo I. This DEC PDP-8 — released in the mid-1960s and affectionately known as the "Straight 8" — is on display at London Science Museum *(Courtesy of Kirk1013 via Wikimedia Commons)*

Cross Streets: Microcomputers and Amateur Radio, Part II

A Checkered History of Getting With the Programs — Computer, That Is

By Cory GB Sickles, WA3UVV
<wa3uvv@gmail.com>

In Part I of *Cross Streets: Microcomputers and Amateur Radio,* we got a quick overview of early computers, the birth of microprocessors and some idea of how microcomputers came to be.

The hearty souls who designed and put together early systems — some from scratch — did it for the fun of experimentation and to learn something new. Many of them were radio amateurs.

However, as with many things, this world sometimes morphed to a point where it was to become less about something you *spent* money on and more about something you could *make* money from. In order to get there, you had to get to a point where things became *practical.*

In the early days of microcomputers, many of us were just happy to be able to add up a couple numbers or blink some lights. As memory — RAM and ROM — became less expensive, along with simpler and speedier ways to save data, those programs became larger.

While programming in machine language was possible, it wasn't much fun for anything larger than a few hundred bytes. Assemblers and editors made things simpler and allowed for easy flexibility in adding or modifying code. But the availability of high-level interpretive languages like BASIC is what fueled an explosion of software development and sharing.

Following a meeting with Charles Tandy and some other key executives, it was decided to move forward with production of the renowned TRS-80. A run of 3,500 was a "safety" number, since there were about that many RadioShack® stores in the country at the time. "If memory serves, the initial showing of the TRS-80 resulted in that many orders almost overnight," writes WA3UVV. *(Courtesy of Okona via Wikimedia Commons)*

Getting Down to BASIC

If you knew someone with the same processor on the same system, then sharing code between you was practical. But if *you* wrote something for a 6502 on an OSI-500 and I had an 8080-based IMSAI, then there was no sharing.

Your chip's architecture, op-codes, and I/O assignments weren't even going to be close to mine. If we both used the same processor, but on different systems, then there was still a problem, albeit a smaller one. It suffices to say, the code written for a 6502-based KIM wasn't going to be *plug and play* on a 6502-based Apple I.

But with a common language like BASIC on each of our systems, program "portability" became more of a reality. Sure, there might be a few differences in optional commands from version to version, but the core routines of PRINT, GOTO, LET, FOR/NEXT, and so on, remained the same.

Perhaps no one was credited for getting BASIC on more systems than Tom Pitman — especially those with just a few K of memory, **Photo A**.

At this point I should also mention Dick Whipple, Bob Albrecht, John Arnold, Dennis Allison, the Homebrew Computer Club, **Photo B**, and "Dr. Dobbs Journal of Tiny BASIC Calisthenics and Orthodontia," People's Computer Company, and many more — all were significant to

Photo A. This classic white-on-black video display shows BASIC programming, a computer language that enjoyed great popularity in the early days of microcomputers. *(Courtesy of Wikimedia Commons. For picture credit, visit <http://bit.ly/XFA9hk>)*

the concept, growth, and near-explosion of affordable BASIC on micros. (Yes, as with any list, I know I'm leaving off names and regret that in advance) Three other names I should mention are Paul Allen, Bill Gates, and Microsoft, **Photos C** and **D**.

Enter: Tiny BASIC

While Microsoft had nothing directly to do with Tiny BASIC, it did have something to do with it indirectly. Microsoft was charging $150 for its version of BASIC. That was a lot of money then and hobbyists wanted less expensive — or free — alternatives. The introduction of Tiny BASIC represented that, without software piracy, a problem that started long before the microprocessor was invented.

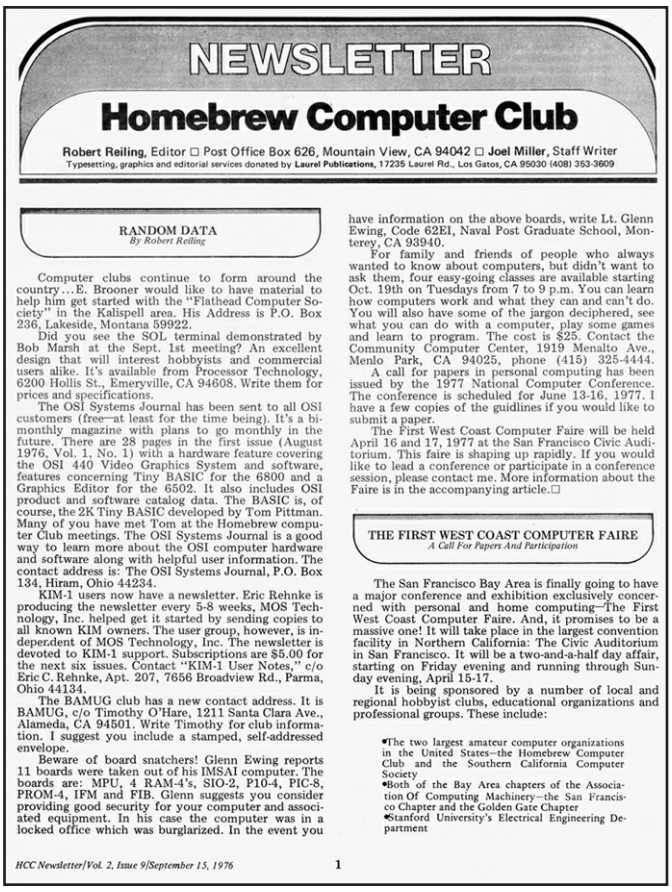
As Tiny BASIC developed, you could get versions for just about every popular system on various forms of media (paper tape, Kansas City Standard cassette tape, **Photo E**, and so on) for very reasonable prices.

If you were going to be able to type in long programs and quickly debug them, then you were going to want something better than toggle switches. Terminals became even more popular and increased sales resulted in lower costs and more options. The American Standard Code for Information Interchange (ASCII) is more recent than early computers. In fact, a popular terminal for systems as recent as the early DEC PDP series was a Teletype Model 28 ASR, **Photo F**. *Why?* Because that's what was available.

Learning the Language

With a Baudot or Murray code teleprinter, the five-level code offered only 32 possible (2^5) characters. Designers extended this with the FIGS and LTRS shift codes, but still, it didn't offer a character set that was as expansive as what was becoming necessary.

The first high-level language I learned was FORTRAN (FORmula TRANslator) and that was on a Honeywell 200, which was an IBM 1401 clone. ***(NOTE:** The first IBM compatible? – WA3UVV.)*



NEWSLETTER

Homebrew Computer Club

Robert Reiling, Editor □ Post Office Box 626, Mountain View, CA 94042 □ Joel Miller, Staff Writer

Typesetting, graphics and editorial services donated by Laurel Publications, 17235 Laurel Rd., Los Gatos, CA 95030 (408) 353-3609

RANDOM DATA

By Robert Reiling

Computer clubs continue to form around the country…E. Brooner would like to have material to help him get started with the "Flathead Computer Society" in the Kalispell area. His Address is P.O. Box 236, Lakeside, Montana 59922.

Did you see the SOL terminal demonstrated by Bob Marsh at the Sept. 1st meeting? An excellent design that will interest hobbyists and commercial users alike. It's available from Processor Technology, 6200 Hollis St., Emeryville, CA 94608. Write them for prices and specifications.

The OSI Systems Journal has been sent to all OSI customers (free—at least for the time being). It's a bimonthly magazine with plans to go monthly in the future. There are 28 pages in the first issue (August 1976, Vol. 1, No. 1) with a hardware feature covering the OSI 440 Video Graphics System and software, features concerning Tiny BASIC for the 6800 and a Graphics Editor for the 6502. It also includes OSI product and software catalog data. The BASIC is, of course, the 2K Tiny BASIC developed by Tom Pittman. Many of you have met Tom at the Homebrew computer Club meetings. The OSI Systems Journal is a good way to learn more about the OSI computer hardware and software along with helpful user information. The contact address is: The OSI Systems Journal, P.O. Box 134, Hiram, Ohio 44234.

KIM-1 users now have a newsletter. Eric Rehnke is producing the newsletter every 5-8 weeks, MOS Technology, Inc. helped get it started by sending copies to all known KIM owners. The user group, however, is independent of MOS Technology, Inc. The newsletter is devoted to KIM-1 support. Subscriptions are $5.00 for the next six issues. Contact "KIM-1 User Notes," c/o Eric C. Rehnke, Apt. 207, 7656 Broadview Rd., Parma, Ohio 44134.

The BAMUG club has a new contact address. It is BAMUG, c/o Timothy O'Hare, 1211 Santa Clara Ave., Alameda, CA 94501. Write Timothy for club information. I suggest you include a stamped, self-addressed envelope.

Beware of board snatchers! Glenn Ewing reports 11 boards were taken out of his IMSAI computer. The boards are: MPU, 4 RAM-4's, SIO-2, P10-4, PIC-8, PROM-4, IFM and FIB. Glenn suggests you consider providing good security for your computer and associated equipment. In his case the computer was in a locked office which was burglarized. In the event you have information on the above boards, write Lt. Glenn Ewing, Code 62EI, Naval Post Graduate School, Monterey, CA 93940.

For family and friends of people who always wanted to know about computers, but didn't want to ask them, four easy-going classes are available starting Oct. 19th on Tuesdays from 7 to 9 p.m. You can learn how computers work and what they can and can't do. You will also have some of the jargon deciphered, see what you can do with a computer, play some games and learn to program. The cost is $25. Contact the Community Computer Center, 1919 Menalto Ave., Menlo Park, CA 94025, phone (415) 325-4444.

A call for papers in personal computing has been issued by the 1977 National Computer Conference. The conference is scheduled for June 13-16, 1977. I have a few copies of the guidlines if you would like to submit a paper.

The First West Coast Computer Faire will be held April 16 and 17, 1977 at the San Francisco Civic Auditorium. This faire is shaping up rapidly. If you would like to lead a conference or participate in a conference session, please contact me. More information about the Faire is in the accompanying article.□

THE FIRST WEST COAST COMPUTER FAIRE

A Call For Papers And Participation

The San Francisco Bay Area is finally going to have a major conference and exhibition exclusively concerned with personal and home computing—The First West Coast Computer Faire. And, it promises to be a massive one! It will take place in the largest convention facility in Northern California: The Civic Auditorium in San Francisco. It will be a two-and-a-half day affair, starting on Friday evening and running through Sunday evening, April 15-17.

It is being sponsored by a number of local and regional hobbyist clubs, educational organizations and professional groups. These include:

- The two largest amateur computer organizations in the United States—the Homebrew Computer Club and the Southern California Computer Society
- Both of the Bay Area chapters of the Association Of Computing Machinery—the San Francisco Chapter and the Golden Gate Chapter
- Stanford University's Electrical Engineering Department

HCC Newsletter/Vol. 2, Issue 9/September 15, 1976 1

Photo B. Homebrew Computer Club, which produced a newsletter edited by Robert Reiling, was one of the key players in the "growth and near-explosion of affordable BASIC on micros," writes WA3UVV. This edition is from September 1976. *(Courtesy of Wikimedia Commons)*

When learning the language, I wondered why certain mathematical comparatives were not just typed in, (=, <, >, and so on), since they were on the keyboard of the Teletype Model 33 ASR I was using.

After researching it a bit (we used public libraries, instead of search engines, then) I realized that due to the age of FORTRAN, the language was designed to work better with the peripherals of the day. So, certain things were expressed with "normal" letters, as allowed by the five-level code of the day. Here are some examples:

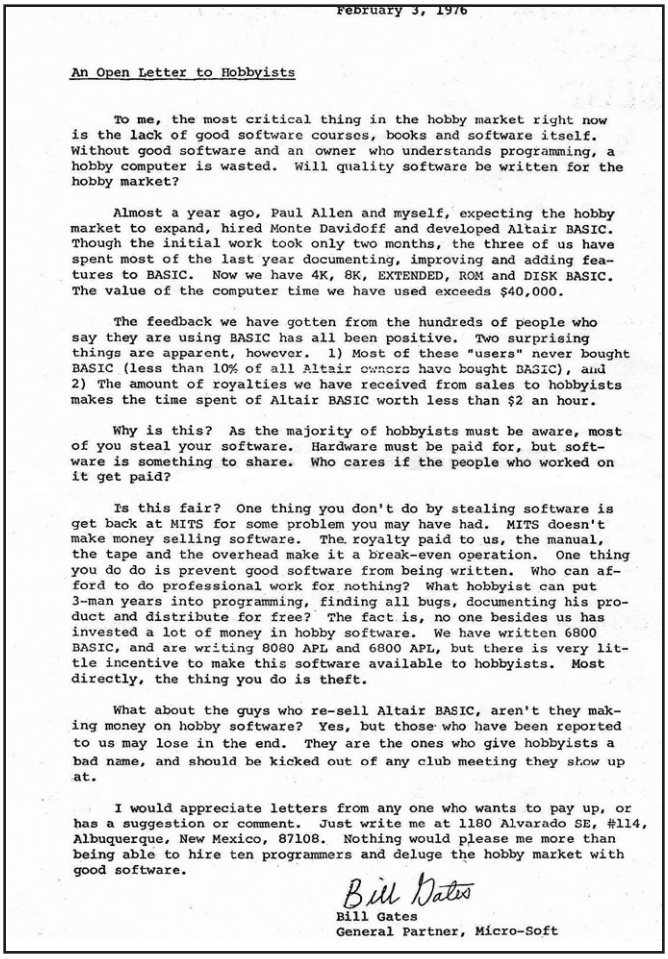
- .EQ. = Equals
- .LT. < Less Than
- .GT. > Greater Than
- .LE. <= Less Than or Equals
- .GE. >= Greater Than or Equals

Hamfest Treasures

My first set of terminals came from a trip to a local hamfest. I purchased a pair of Teletype Model 2Bs and a Model 15. The Model 15, **Photo G**, cost me $15 and I got both Model 2Bs — and a water bottle — for an additional $4.

If you're wondering about the 2B and the water bottle, it printed on a thin strip of paper, which was run through a moistening pad on a bottle, then cut to length and glued onto a piece of paper, **Photo H**. Watch an old movie scene where someone

Photo C. "The MITS Altair 8800 computer was the first commercially successful home computer," according to an entry on Wikimedia Commons. "Paul Allen and Bill Gates wrote Altair BASIC and started Microsoft. This advertisement appeared in *BYTE, Popular Electronics, Radio-Electronics* and other magazines in May 1976."
(Courtesy of Wikimedia Commons)



February 3, 1976

An Open Letter to Hobbyists

To me, the most critical thing in the hobby market right now is the lack of good software courses, books and software itself. Without good software and an owner who understands programming, a hobby computer is wasted. Will quality software be written for the hobby market?

Almost a year ago, Paul Allen and myself, expecting the hobby market to expand, hired Monte Davidoff and developed Altair BASIC. Though the initial work took only two months, the three of us have spent most of the last year documenting, improving and adding features to BASIC. Now we have 4K, 8K, EXTENDED, ROM and DISK BASIC. The value of the computer time we have used exceeds $40,000.

The feedback we have gotten from the hundreds of people who say they are using BASIC has all been positive. Two surprising things are apparent, however. 1) Most of these "users" never bought BASIC (less than 10% of all Altair owners have bought BASIC), and 2) The amount of royalties we have received from sales to hobbyists makes the time spent of Altair BASIC worth less than $2 an hour.

Why is this? As the majority of hobbyists must be aware, most of you steal your software. Hardware must be paid for, but software is something to share. Who cares if the people who worked on it get paid?

Is this fair? One thing you don't do by stealing software is get back at MITS for some problem you may have had. MITS doesn't make money selling software. The royalty paid to us, the manual, the tape and the overhead make it a break-even operation. One thing you do do is prevent good software from being written. Who can afford to do professional work for nothing? What hobbyist can put 3-man years into programming, finding all bugs, documenting his product and distribute for free? The fact is, no one besides us has invested a lot of money in hobby software. We have written 6800 BASIC, and are writing 8080 APL and 6800 APL, but there is very little incentive to make this software available to hobbyists. Most directly, the thing you do is theft.

What about the guys who re-sell Altair BASIC, aren't they making money on hobby software? Yes, but those who have been reported to us may lose in the end. They are the ones who give hobbyists a bad name, and should be kicked out of any club meeting they show up at.

I would appreciate letters from any one who wants to pay up, or has a suggestion or comment. Just write me at 1180 Alvarado SE, #114, Albuquerque, New Mexico, 87108. Nothing would please me more than being able to hire ten programmers and deluge the hobby market with good software.

Bill Gates

Bill Gates
General Partner, Micro-Soft

Photo D. In an open letter dated February 3, 1976 to the Homebrew Computer Club, Bill Gates raised the issue of software piracy — the shape of things to come.
(Courtesy of Wikimedia Commons. For an enlarged view, visit <http://bit.ly/QdK4Zy>.)

gets a telegram that the camera frames close-up and you'll see the end result.

Compare that to a Model 33 selling for more than $700 and you'll see why this was so attractive. Yes, I had to write some assembly language code to talk to the teleprinters. I tried a 2B first, in case I messed something up. And I had to adapt a FORTRAN-inspired way to display and recognize special characters, but after a few hours of experimentation, I had a working terminal!

In short order, I burned an EPROM with the driver, so it was easy to load the driver each time the computer was powered on. Eventually, I picked up a paper tape reader and punch. Then, I wrote some additional software to save my programs on five-level tape, using a hex code format. It was slow, but cheap and fairly reliable. *(**IN DEPTH:** See Note 1 at <http://bit.ly/clUIPC>.)*

As time passed, I replicated this idea across several single board computers I owned, plus a TRS-80 and Apple II.

Microcomputers, RTTY, and the Radio Amateur

Speaking of Teletypes, one of the first ham uses of microcomputers was with RTTY. No longer did you have to have a set of "brag tapes" stored on clotheslines or in pie tins. Now, you could just let the computer send the data, effortlessly and with less noise.

Storing entire exchanges became a reality, as did early attempts at automatic logging. As a CW identification was required at the time, the computer could do that, too. It could keep track of the time, and knew when it was time to ID, as well.

The creation of mailboxes was made possible by micros and some well-written software. Computers also made it possible to have modes like AMTOR, PACTOR, and — later of course — Packet and APRS.

If you weren't a fan of RTTY, or wanted something faster, there was always CW. That's CW with speeds in the hundreds of words per minute, mind you. While many of us were happy to program our computers to send and receive CW at "normal" speeds, some pushed the envelope with hyper-speed code and even exchanged software that way.

Other Ham Applications

Slow-Scan TV (SSTV) benefited from computers in the shack, as well. With ever-decreasing memory costs and the

Photo E. Introduced in May 1976 for $80, the Southwest Technical Products Corporation (SWTPC) AC-30 tape interface for early personal computers is shown with a Sankyo ST-30 cassette recorder, a SWTPC 8K BASIC tape and a TSC 6800 Assembler tape. The AC-30 was designed to meet the proposed Kansas City Standard data format. *(Courtesy of Michael Holley via Wikimedia Commons)*

growth of storage options, your computer could send and receive SSTV, plus store pictures for later or allow you to batch a number together, thus allowing for short programs of sorts, **Photo I**. Memory could be allocated for frame buffers that allowed SSTV pictures to be displayed on a standard TV screen. All this by virtue of some simple (or not so simple, if you'd rather) homebrew interfaces.

No longer did you need to have the bright lights on when you sent a picture, only to have to dim the room to see what was being received.

The growing interest in satellite operation was further enhanced when software was written to calculate passes and tracking information. It became "way cool" however, when you added capabilities to the software, built an interface, and controlled the azimuth/elevation rotors — thus freeing up your hands from that task, **Photo J**.

Are you still using a paper log, other than when operating mobile or portable? If so, you are in the vast minority. Logging software was one of the first examples of "hobby begets income" products for the amateur radio operator that didn't want to write their own.

Developing Operating Systems

While all this was developing, another set of standards was emerging. With the lowering costs of disk drives, operating systems (OS) came into vogue. Just as a standard language had allowed growth, so would operating systems.

While each computer system manufacturer seemed to offer their own, one in particular made inroads and more importantly, was designed to be portable across a variety of systems. Gary Kildall, a developer at Intel, developed a nicely-featured OS and set of utilities. Some may see it as a substantial improvement over the ISIS OS.

As what you will discover to be a recurring story, Kildall went to his employers with his project to offer it to them. He was basically told that Intel was a hardware company and had no interest in selling software. He was thus free to pursue his work without any constraints by his employer. He started a small company known as Digital Research. The OS was named CP/M, for Control Program for Microcomputers.

Photo F. Housed under the East Wing of the White House, the Washington-Moscow Hotline in the WHCA-communications center used Model 28 ASR and Siemens T63 SU12 teletype equipment, each connected to an ETCR-RM II encryption device. *(Courtesy of Wikimedia Commons)*

Photo G. The TG-7-B teletype machine was a military version of the Model 15. *(Courtesy of the Computer History Museum <http://bit.ly/UpluzW>, via Wikimedia Commons)*

Photo H. A YouTube video titled Telephone and Telegraph shows the intricacies of teletype systems (***WATCH:*** *At <http://bit.ly/UDpFhE>.) (Internet screen grab)*

Photo I. Here is video of a 20-meter, slow-scan TV contact between KG4I in Birmingham, Alabama and N2BJW in Elmira, New York that took place on the afternoon of June 14, 1980. <http://bit.ly/SVCmzn>. *(Internet screen grab)*

CP/M was initially available on 8-inch floppy discs, the *drive du jour* of the time. Single-Sided, Single-Density (SDSS, a.k.a. IBM 3740) format disks could store 241K of data. It doesn't sound like much today, but this was a reasonable amount of information for a time when systems maxed out at 64K of memory.

While floppy drives had existed well before the microprocessor, the latter caused the refinement of the former, resulting in — you guessed it — lower prices, wider variety and increased features. *(**NOTE:** Isn't capitalism wonderful? – WA3UVV)*

Shugart Associates, created by Alan Shugart — formerly of IBM and Memorex — introduced a line of 8-inch drives and CP/M seemed the perfect thing to stick into one.

While CP/M was designed to work only on 8080 and Z-80 systems, with just minor "tweaks" there were a lot of systems out there. Having a common OS available meant there were more to come. Having a reasonable amount of storage available meant more complex software could be written that could carry out business applications.

Having such software available encouraged the sales of even more systems and all this encouraged more product development — both hardware and software.

As 8-inch disk development continued, SDSS format remained a common standard, but Double-Sided, Double-Density (DSDD) formats became available. Capacities of 448K, 566K and upwards of the megabyte range were offered, increasing the capabilities of microcomputers and what they could do. At the same time, things continued to get smaller and the 5-1/4-inch Minifloppy drive was introduced. Originally, it offered less than 90K of SSSD formatted space, but that increased in time.

*(**AN OS FOOTNOTE:** While CP/M offered portability across the 8080 and Z-80 series chips, another OS was eventually developed in the 1970s to allow portability across systems using different processors. UCSD Pascal was an OS — with a common language — that would allow the same application to run on an 8080, 6800, LSI-11, you name it. Each system had an interpretive core or kernel that was machine-specific. That meant your programs executed equally slowly on all*

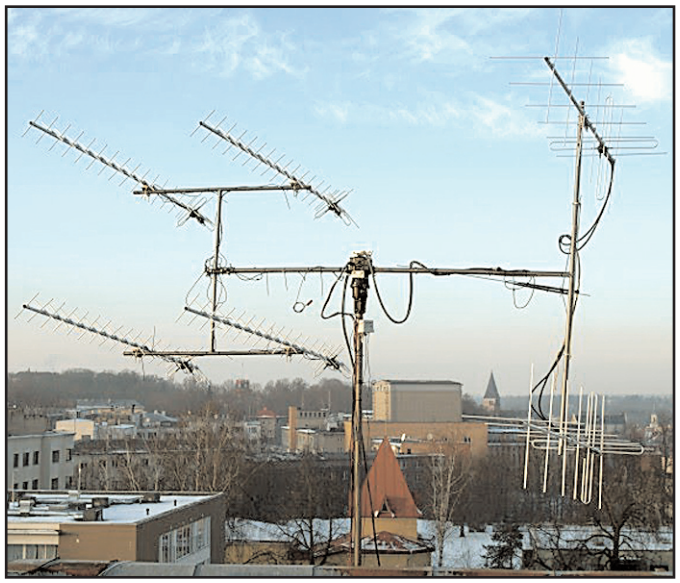


Photo J. Amateur radio satellite operation "became *way cool* when you added capabilities to the software, built an interface, and controlled the azimuth/elevation rotors — thus freeing up your hands from that task," writes WA3UVV. *(Courtesy of ESTCube/Urmas Kvell via Wikimedia Commons)*

supported systems. Still, it was a step in the right direction. – WA3UVV)

Expanding Parallel Universes

As many things happen concurrently, there was another important series of developments in hardware — the *all in one* computer. Having developed video games and having a genius-level gift for reducing the parts count in a design, Steve Wozniak, **Photo K**, ex-WA6BND, developed a microcomputer later dubbed the Apple I.

Steve showed his design to his bosses at Hewlett Packard. *You know where this is going, right?* The suits at HP dismissed the opportunity, stating HP was a "big" computer company and wasn't interested in the "little" microcomputer market.

Using a 6502 processor — selected because of low cost — the Apple I was the first micro not requiring an external terminal, **Photo L**. In addition, it used a keyboard that could be placed in the same enclosure and a TV for the display — *in color!* It may not sound like much right now, but this was a major development.

Wozniak's design incorporated BASIC in ROM and was ready to go when you turned it on. The Apple II soon followed. In one rather nice-looking case, it had the computer, room for 48K or RAM, BASIC and a monitor in ROM, plus expansion slots, power supply, and video output. Game controller I/O and a few more surprises were in there, too. It really raised the bar.

Soon to appear would be Chuck Peddle's PET 2001. This design incorporated the video monitor in the enclosure, as well as a cassette recorder. It also had a horrible, atrocious, "Chiclet" keyboard with all sorts of special graphics symbols. While Apple chose to sell its systems through the emerging network of computer stores, the PET was distributed through department store retailers like Sears.

Meanwhile, in an isolated room at the end of the hall, somewhere in Texas, an engineer was tempted to leave his job at National Semiconductor. Steve Leininger was certainly familiar with digital electronics, having designed National's SC/MP microprocessor. Through a chance meeting, then another, he was offered a position at the Tandy Corporation — parent company of RadioShack®. Leininger was initially given non-computer projects to work on. But with the abrupt end of the CB feeding frenzy and the not-yet mature VCR market, he was allowed to focus his attentions on a computer system.

After toiling for months on end he finally had a wire-wrapped design that worked consistently well enough to demonstrate to Charles Tandy and some other key executives. At the end of the meeting, it was decided to move forward with production, albeit only 3,500. That was a "safety" number, since there were about that many stores in the country at the time and it was felt that if the computer didn't sell, it could be used for "something" in the stores. If memory serves, the initial showing of the TRS-80 resulted in that many orders, almost overnight.

That kind of demand could kill a smaller company. But Tandy had deep pockets and was able to ramp up quickly to meet demand. Still, those who placed initial orders might wait between three to four months for delivery.

While the TRS-80 wasn't a "better" system than Apple's, it did have two things going for it. One was the resources of Radio Shack® to market the system and influence developers. The other was a book.

Excellently written by David Lien, WY6T, the *TRS-80 Level I Users Manual* was key to the success of that computer. Lien's style introduced the concepts of a computer, programming, and the functions of each command of the BASIC language in terms that the average and well-below-average person could understand.

I could actually go on for several pages of praise for the way the book was written and how significant it was as both a sales tool and a means to keep customers from calling the store with simple, repetitive support questions.

Career Opportunities

Yes, I worked at RadioShack® in those days. I sold lots of computers and made lots of money doing it. Further, it enabled me to start a small vertical market software firm, with clients like dentists and insurance agents. When I wasn't working in the store, chasing girls, or driving too fast, I was sitting in the basement hammering out code and drawing flow charts.

The PET eventually got a "real" keyboard. The Apple II went on to be the first microcomputer to outsell the DEC PDP-8. RadioShack®, bolstered by TRS-80 sales, converted stores throughout the country into RadioShack® Computer Centers (Tandy Computer Centers were a short-lived experiment). Part-time programmers almost fell over themselves, producing software for their favorite systems, still mostly on cassette.

Pioneering Spirit

Note that none of these systems used the S-100 or SS-50 buss. They each forged ahead with their own. While Apple actually had expansion slots within the computer, Commodore's (formerly MOS Technology) PET used the IEEE-488 (or GPIB) buss, originally designed for instrumentation control and communications.

Radio Shack®'s design was the most convoluted. You could expand the computer-in-a-keyboard to 16K and Level II BASIC. To go further, you needed to purchase an Expansion Chassis, which sat under the monitor, connected to the computer via a semi-shielded 40-pin ribbon cable.

Photo K. Steve Wozniak, ex-WA6BND, developed a microcomputer later dubbed the Apple I. And the rest, as they say, is computer history. *(Courtesy of Wikimedia Commons)*

The Expansion Chassis retailed for $300 without any additional memory! That was just the price of the box, with additional electronics inside. With another $30 purchase you could have a parallel printer cable. Inside the Expansion Chassis, up to 32K more of memory could be added — by an authorized technician. Otherwise, you voided your warranty. RadioShack® used a sealant on one of the screws, to show if the box had been tampered with. *(**NOTE:** If you were very careful, you could still remove that screw and re-form the sealant when you were done with your own upgrade. Not that I ever did that, of course. – WA3UVV)*

With the Expansion Chassis, you could add up to four 5-1/4-inch mini-floppy drives at $500 each, plus an RS-232C serial interface for an additional $150. Initially, a regular-paper friction-feed printer — private-labeled Centronics 779 — sold for $1,300. So, once you upgraded your initial $599.99 entry-level system to something you could do business with, you had ponied up about $3,800! Of course if you were still using a Model 15 as your printer, *like some people who shall remain nameless,* you got to save $1,300 for other things.

Radio Amateurs vs. Computer RFI

As you might imagine, all those exposed ribbon cables coming out of plastic enclosures with circuits generating square waves at 4 MHz and under, created a bit of an RF issue. This was another thing that early computer hobbyists who were also radio amateurs had to contend with.

While some systems like the Apple II generated a lot less, the TRS-80 was notorious for a lack of shielding. Trying to use one near an amplifier would turn the screen white.

In return, the processor noise could wipe out all but the strongest signals. RadioShack®'s "fix" for the problem was to seriously downgrade the processor speed to — if memory serves — 500 kHz. *Some fix.*

My solution was to line the inside of everything with aluminum foil and place ferrite chokes on all the exposed cables.

The fact was that just about every computer system on the market had always generated that kind of noise. What was different now was the substantial population of computers. As sales continued to climb, something had to be done.

The FCC stepped in and had a new set of compliance standards in place by the early '80s. There was one classification for "home" computers and another, less stringent, classification for "business" computers. *(**QUESTION:** Guess which classification many of the formerly marketed-for-home-use systems were now marketed as? – WA3UVV)*

Look Ahead

While this catches us up to changes in the microcomputer world from the introduction of microprocessor to the solid entrenchment in the office, home, and shack at the end of the 1970s, there was something looming on the horizon — a *super-secret project* taking place in Boca Raton, Florida — that would change it all.

Fortunes were about to be made — and opportunities lost — once again. More about that next time in Part III. **–** *WA3UVV*

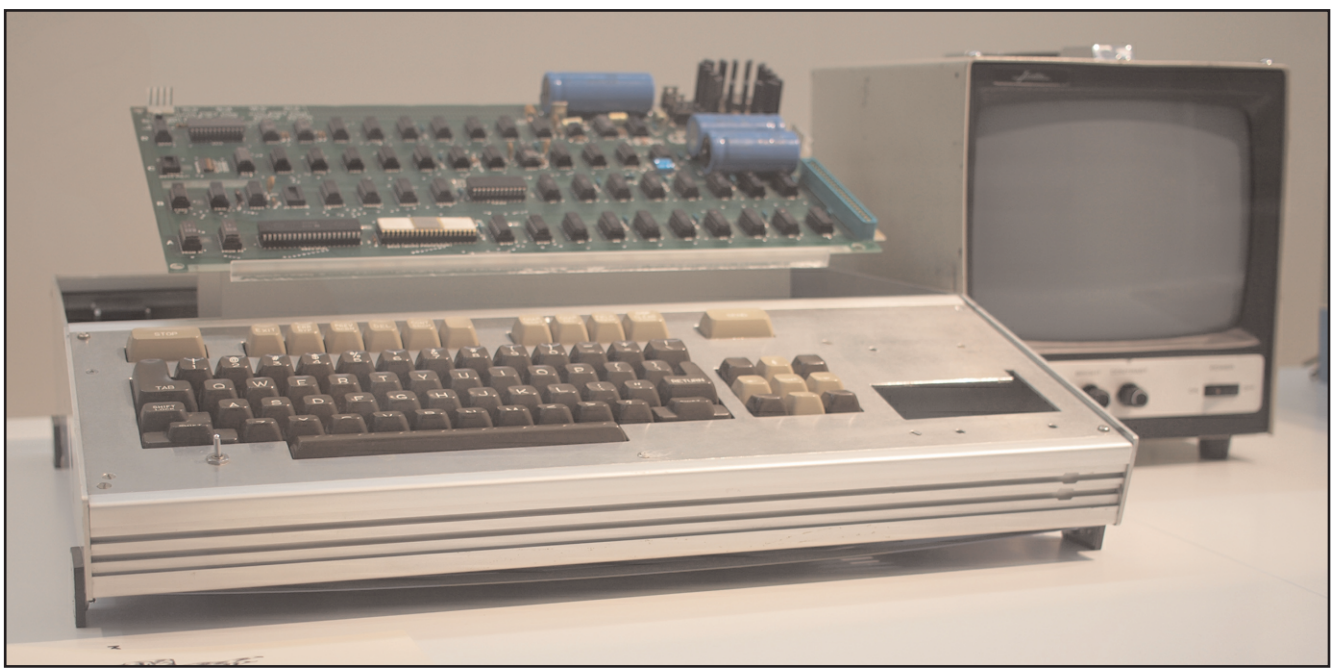


Photo L. A classic Apple I computer is on display at Museum für Kommunikation Bern in Switzerland. *(Courtesy of Tomislav Medak via Wikimedia Commons)*